



Kondisi

Pada C++



Atang Supena

Kondisi

1. PERNYATAAN KONDISI (CONDITIONAL EXPRESSION)

Pertanyaan Kondisi dibagi menjadi,

1.1. Pernyataan *if*

Digunakan dalam pengambilan keputusan

Bentuk umum:

```
if(kondisi) pernyataan1 ;  
    else pernyataan2;
```

Pernyataan1 dilaksanakan jika dan hanya jika kondisi yang diinginkan terpenuhi, jika tidak, lakukan pernyataan2.

Jika anda tidak mempergunakan pernyataan *else* program tidak akan error, namun jika anda mempergunakan pernyataan *else* tanpa didahului pernyataan *if*, maka program akan error.

Jika pernyataan1 atau pernyataan2 hanya terdiri dari satu baris, maka tanda { } tidak diperlukan, namun jika lebih maka diperlukan.

Bentuknya menjadi :

```
if(kondisi)  
{  
    pernyataan1;  
    pernyataan1a;  
    pernyataan1b;  
}  
else  
{  
    pernyataan2;  
    pernyataan2a;  
    pernyataan2b;  
}
```

Contoh :

```
#include <iostream.h>

void main()
{
    int m = 166;
    if(m == 0)cout<<"Nilainya sama dengan nol\n";
    else
    {
        cout<<"Nilainya tidak sama dengan nol\n";
        cout<<"Nilainya sama dengan "<<m<<endl;
    }
}
```

Selain dari if ... else, juga dikenal bentuk if ... else if. Adapun perbedaannya diilustrasikan oleh dua contoh dibawah ini.

Contoh 1 :

```
#include <iostream.h>

void main()
{
    int m = 166;

    if(m > 1000) cout<<m<<" lebih besar dari 1000\n";
    if(m > 100) cout<<m<<" lebih besar dari 100\n";
    if(m > 10) cout<<m<<" lebih besar dari 10\n";
}
```

Keluarannya :

166 lebih besar dari 100

166 lebih besar dari 10

Contoh 2 :

```
#include <iostream.h>

void main()
{
    int m = 166;

    if(m > 1000) cout<<m<<" lebih besar dari 1000\n";
    else if(m > 100) cout<<m<<" lebih besar dari 100\n";
    else if(m > 10) cout<<m<<" lebih besar dari 10\n";
}
```

Keluarannya :

166 lebih besar dari 100

Mengapa ? Karena contoh 2 sama saja jika ditulis seperti dibawah ini

```
#include <iostream.h>

void main()
{
    int m = 166;

    if(m > 1000) cout<<m<<" lebih besar dari
    1000\n"; else
    {
        if(m > 100) cout<<m<<" lebih besar dari 100\n";
        else if(m > 10) cout<<m<<" lebih besar dari 10\n";
    }
}
```

Contoh diatas disebut juga *nested conditional*

4.6.2 Pernyataan *switch*

Pernyataan *if...else if* jamak dapat dibangun dengan pernyataan *switch*. Bentuk umumnya adalah sebagai berikut.

```
switch(ekspresi)
{
    case konstanta1 : pernyataan1;
    case konstanta2 : pernyataan2;
    case konstanta3 : pernyataan3;
        :
        :
    case konstantaN : pernyataanN;
    default : pernyataanlain;
}
```

Hal – hal yang perlu diperhatikan adalah :

1. Dibelakang keyword case harus diikuti oleh sebuah konstanta, tidak boleh diikuti oleh ekspresi ataupun variable.
2. Konstanta yang digunakan bertipe int atau char
3. Jika bentuknya seperti diatas maka apabila *ekspresi* sesuai dengan konstanta2 maka pernyataan2, pernyataan3 sampai dengan pernyataanlain dieksekusi. Untuk mencegah hal tersebut, gunakan keyword *break*;. Jika keyword *break* digunakan maka setelah pernyataan2 dieksekusi program langsung keluar dari pernyataan *switch*. Selain digunakan dalam *switch*, keyword *break* banyak digunakan untuk keluar dari pernyataan yang berulang (looping).
4. pernyataanlain dieksekusi jika konstanta1 sampai konstantaN tidak ada yang memenuhi *ekspresi*.

Contoh :

```
// Program untuk melihat nilai akhir test
// Nilai A jika nilai diatas 80, B jika 70<= nilai <80
// C jika 50<= nilai <70, D jika 30<=nilai <50
// E jika nilai < 30
#include <iostream.h>

void main()
{
    int nilai;
    cout<<"Masukkan nilai test : ";
    cin>>nilai;
    switch(nilai/10)
    {
        case 10:
        case 9:
        case 8:cout<<'A'<<endl;break;
        case 7:cout<<'B'<<endl;break;
        case 6:
        case 5:cout<<'C'<<endl;break;
        case 4:
        case 3:cout<<'D'<<endl;break;
        case 2:
        case 1:
        case 0:cout<<'E'<<endl;break;
        default:cout<<"Salah, nilai diluar jangkauan.\n";
    }
}
```



Keluaran :

Masukkan nilai test : *45*

D

Masukkan nilai test : *450*

Salah, nilai diluar jangkauan.

Masukkan nilai test : *nilai_test*

Salah, nilai diluar jangkauan.

Ket : 45, 450 dan *nilai_test* adalah hasil input dari user



-
1. Deitel & Deitel, C How to Program 3rd Edition, Prentice Hall, New Jersey, 2001
 2. Jogiyanto, Konsep Dasar Pemrograman Bahasa C, Andi Offset, Yogyakarta, 1993
 3. Moh. Sjukani, 2009, Algoritma & Struktur Data 1 dengan C, C++ dan Java, Edisi 5, Mitra Wacana Media, Jakarta.
 4. Thompson Susabda Ngoen, Pengantar Algoritma dengan Bahasa C, Salemba Teknik, Jakarta, 2004.

